The Virtual Composer

J____E____

Abstract

When using software, namely video games, there is often background music for certain areas – menus, areas and locations inside of games, etc. Not only can this music be expensive to buy, but also becomes tedious if only a small amount of it is bought. However, if companies were only required to buy a simple program that would constantly make them new music, keeping the tunes of the software fresh, these problems could be solved. But would people dislike computer-generated music?

Thus, it was hypothesized that a clip of music composed by a program would be virtually undistinguishable, at least on the first listen, from one composed by a human. The program generated a chord progression, which it then built a melody over using musical knowledge programmed into it. A survey resulted that while less than 25% of all people were likely to find the results unpleasant or annoying, the same could not be said for finding it un-musical.

Introduction

Background music has long been an effective way to build comfort or mood. It has spilled over into the modern world in many places, including virtual products. Whether it is playing on a menu screen for On Demand TV or echoing around a tavern in a fantasy video game, music is still relied upon for the same things it always has: comfort and mood. Thinking is often easier over background music. Immersion, a vital part in the success of most video games, settles upon the consumer more effectively.

Yet there are problems with obtaining music. It is expensive, as composers often want large payments for well-crafted music. It is even more expensive if you want to hire real musicians with a multitude of instruments to play it, as opposed to a computer or a studio-engineer with a synthesizer. The money spent on this could be going towards the actual product, instead of supplements to the product. Thus companies are often discouraged from expanding on this important aspect, and either do it in a lackluster—sometimes outright detrimental—way, or not at all. They may only buy one or two songs, which are played constantly and annoy the user, who simply turns off the sound. Yet, if they could buy a simple, cheap program that would instantly compose new music each time it was accessed by their product, this would be much less of a problem and would provide the same benefits of more expensive music, while diverting less effort and financing away from their main focus. Also, a full score of music will take up much more space on both a computer and CD than a small program, which can pose minor complications or inconveniences.

The focus of this project is not only to build an effective virtual music composer, but also to test whether or not people would accept the music composed by a cheap, simple program. If it is shown that, overall, people have no problem with it, the project can be a building block for a corporate-level product. The program will be written in the Java programming language (using the Bluej Java interface) and Melody Assistant to play the program's output.

Creating different classes made the organization and functionality of the program smoother. Outside of the main composer class, one was created to handle all of the information that pertained to a chord. Each chord used in the program was not a string of notes, but rather an instance of this class, within which the notes of the chord was a small part of information. It was within the Chord class that the method to randomly choose a next chord from its given list was stored. An abstract class called "phrases" was made, under which all other phrases (which were their own class) used as a template. Each class of a phrase knew how to create its own given phrase, returning the phrase when called (after it was given particular notes that it could start on, which was determined from the Chord class it would be played simultaneously with).

After the main "compose" method had been run, a .abc file was created. This file was properly formatted to be read by a dedicated music-program—In this case, Melody Assistant. Once opened with this program, individual notes could be seen and the composition could be exported to a music file with could in turn be played. For bug testing, a simple replacement of ".abc" with ".txt" in the file destination line would create a text file in ABC format which could be read.

The Survey

After the program was deemed complete—that is, once it could successfully output a functional piece of music—a single piece was composed and saved as an .mp3 file. Then, three other mainstream, human-composed pieces were rewritten in a manner that matches the output of the program: A chord progression underneath a single-note melody. Clips from the songs "My Father's Eyes" by Eric Clapton, "Hit That" by The Offspring, "Violin Concerto #1" by Paganini, and "We Didn't Start the Fire" by Billy Joel were used. It is important to note that these songs were chosen because they already matched the single-note-melody, basic-chord-progression style, and were thus not altered substantially by the transition. It was also important that they all be of a similar recording quality, so the computer-generated piece does not stand out so blatantly.

Once all pieces were stored as .mp3 files, which could be played by any MP3 player (in this case, an iPod). They were presented in 30-second clips to participants (35 in total: 31 is a minimum for many statistical tests), who then answered 2 simple questions. They were asked to identify any piece that sounds unpleasant or annoying, and any that sounds "un-musical". Thus, after collecting the data, it can be formulated whether or not the output of the program was successful.

Using a random number generator, the experimental clip was assigned on the survey as "clip 1".

With the results, a 1-proportion z-test was used, using a TI-84+ graphing calculator (although it will work with any graphing calculator at least as new as the TI-83).

Results

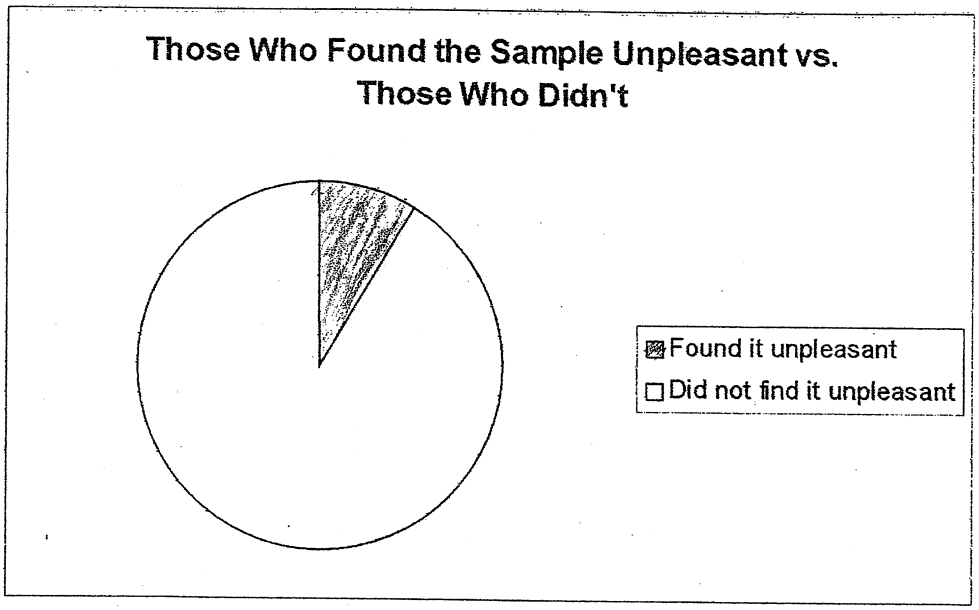
The program was completed successfully. Although errors and snags were encountered, it was negligible and did not prevent the desired result from being achieved. A functional mp3 file was exported, and the other songs were transposed to simpler format as well.

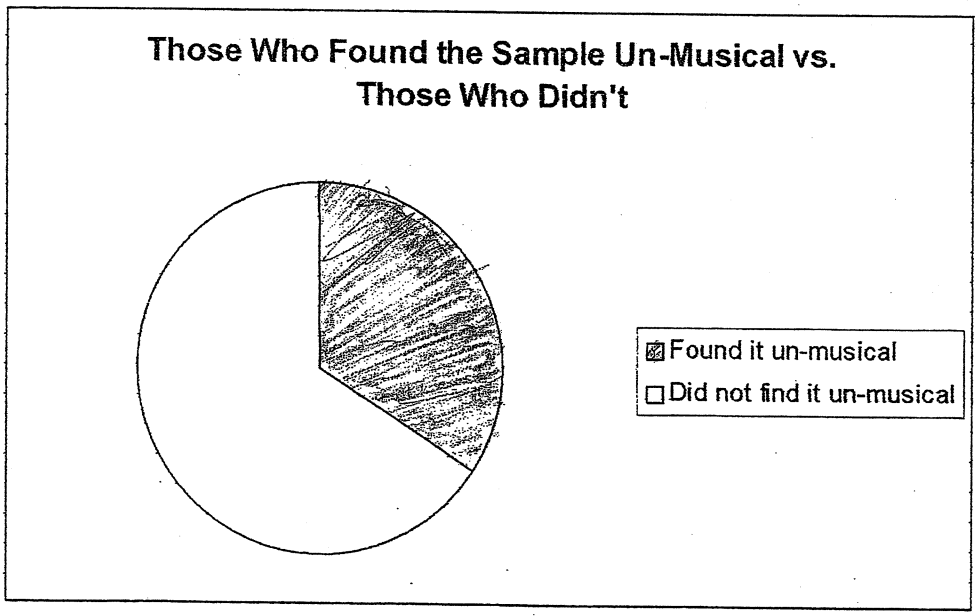
The results of the survey were as follows:

For question one, 2 out of the 35 applicants found the generated clip to be annoying or unpleasant. Question two yielded 12 who found it un-musical. That's 5.71% who found it unpleasant and 34.29% who found it un-musical.

Using a graphing calculator, a 1-prop Z-test of the chance that less than a fourth of all people would find the results of the composer unpleasant resulted in a p-value of .0042. The same test yielded a p-value of .897 for the probability that less than a fourth would find it un-musical.

These p-values must be compared to the significance value chosen earlier. If lower, the notion of less than 25% finding the piece either unpleasant or un-musical is rejected, meaning it is unlikely. Otherwise, it is accepted as being likely.





Found it unpleasant	2
Did not find it unpleasant	32

Found it un-musical	12
Did not find it un-musical	23

Discussion

When checked against the significance value of .1, the p-value of the first question is much lower, meaning that it is very likely that less than 25% of all people find the results of the program unpleasant. The p-value of the second question, however, was much higher than the significance value, meaning that it is very unlikely that 25% or less of the population would find the piece unmusical. Therefore, half of the hypothesis was correct.

The purpose of this experiment was to determine if it was likely that a simple computer composer could take the place of human composers for simple background music in commercial products. If this is to be true, it is absolutely imperative that a virtually-composed piece is not unpleasant to the listener. Not only will it obviously produce discomfort, but may do it to the extent that the consumer will be unsatisfied with the product. While good music may provide a great deal of immersion value for games, bad music can make immersion very difficult. It can disrupt thinking. It is better to have nothing at all than a negative thing. However, this thankfully does not seem to be the case, as it was found very likely that less than a fourth of all people will find it unpleasant. Although a fourth is still a very large portion realistically—especially for a commercial product—for a "rough-draft" of a program, it is suffice to prove that it is very likely for a program to produce pleasant music.

Musicianship is also paramount to the success of good music. There is more to music than just playing notes that sound good together—if it lacks certain character, melody, or feeling, people will not be able to connect with it. Unfortunately, this experiment was not able to prove that a simple program could produce something melodious or otherwise "musical". It is therefore unlikely that such a program could take the place of a main theme song, and certainly not a mainstream stand-alone song.

Although most of the project went off smoothly, some things could have been done more effectively. Vague planning created a few problems and time-sinks when coding the program. If this or a similar project is to be repeated, it will definitely have a more solid plan beneath it before any coding is done. The survey, while mostly effective, could have been more comprehensive and set-up to have more informative results. As it is, all that is learned from the 2nd question (which did not 'pass') is that the portion of the population that would think the piece unmusical is not 25% or lower. The high p-value indicates only that it is very unlikely, not what a more realistic proportion would be (which would be useful to know). The nature of these results makes it difficult to compare the test to other similar tests. Also, the manner in which this experiment could be carried out could vary drastically, meaning that a viable match to compare it with would be difficult to find.

Overall though, it is believed that the experiment as a whole was a success in its goal. The future of a computer composer is not to produce main theme songs or mainstream radio hits, but to generate backround "filler" music. As long as the music is not unpleasant (which the experiment showed it was not), then it doesn't have to be melodious. In fact, melodious music may distract from the desired focus of the product more than other pieces. Judging by this project, a virtual composer seems to have a definite chance of success.

Bibliography

Aloupis, Greg, Thomas Fevens, Tomomi Matsui, Stefan Langerman, Yurai Nunez, Antonio Mesa, David Rappaport, and Godfried Toussaint. "· Algorithms for Computing Geometric Measures of Melodic Similarity." Computer Music Journal 30 (2006): 67-76. School of Computer Science, Louisville. 6 Oct. 2006.

Blood, Brian. "Music Theory & History Online." Dolmetsch Online. 8 Oct. 2006 <http://www.dolmetsch.com/theoryintro.htm>.

Eachus, Peter. "Survey Techniques." University of Salford. University of Salford. 8 Oct. 2006 <http://www.chssc.salford.ac.uk/healthSci/resmeth2000/resmeth/ch2surv.htm>.

Hanninen, Dora A. "· a Theory of Recontextualization in Music: Analyzing Phenomenal Transformations of Reptition." Music Theory Spectrum (2003): 1-39. Research Library, Louisville. 23 Sept. 2006.

Hewlett, Walter B., and Eleanor Selfridge-Field. "Directory of Computer Assisted Research in Musicology." Computing in Musicology 3 (1987): 1-76. Center for Computer Assisted Research in Humanities. 23 Sept. 2006.

McClelland, Ryan. "· Extended Upbeats in the Classical Minuet: Interactions with Hypermeter and Phrase Structure." Music Theory Spectrum 1 (2006): 1-34. Louisville. 23 Sept. 2006.

Narins, Pamela N. "Survey Techniques." Ryerson. Ryerson University. 8 Oct. 2006 <http://www.ryerson.ca/~mjoppe/ResearchProcess/SurveyTechniques.htm>.

Walshaw, Chris. "The Abc Musical Notation Language." The Abc Musical Notation Language. 12 Oct. 2006 <http://abcnotation.org.uk/>. 1 July 2006.

Whitesell, Lloyd. "· Speech-Melody and Primitive Music." Ed. Leon Botstein. The Music Quarterly 88 (2005): 167-203. 23 Sept. 2006.

Zimmerman, Paul, comp. "Chord Theory." Harmony Central. 23 Sept. 2006 <http://www.harmony-central.com/Guitar/chord-and-theory-chart.txt>.

from content area – Physics

analytical / technical sample Gr. 12 2009

The Virtual Composer

J____ E____

title suggests something about purpose but reader must read on to find out what

specialized reader is target audience

use of subheadings helps focus and helps meet needs of audience

Abstract

When using software, namely video games, there is often background music for certain areas – menus, areas and locations inside of games, etc. Not only can this music be expensive to buy, but also becomes tedious if only a small amount of it is bought. However, if companies were only required to buy a simple program that would constantly make them new music, keeping the tunes of the software fresh, these problems could be solved. But would people dislike computer-generated music?

Thus, it was hypothesized that a clip of music composed by a program would be virtually undistinguishable, at least on the first listen, from one composed by a human. The program generated a chord progression, which it then built a melody over using musical knowledge programmed into it. A survey resulted that while less than 25% of all people were likely to find the results unpleasant or annoying, the same could not be said for finding it un-musical.

Background provides context for reader

abstract is a brief overview

authentic purpose – a real question helps produce an authentic purpose

focus { analytical questions – How can we develop a virtual music composer? How will people respond?

Introduction

Background music has long been an effective way to build comfort or mood. It has spilled over into the modern world in many places, including virtual products. Whether it is playing on a menu screen for On Demand TV or echoing around a tavern in a fantasy video game, music is still relied upon for the same things it always has: comfort and mood. Thinking is often easier over background music. Immersion, a vital part in the success of most video games, settles upon the consumer more effectively.

intro – provides background to problem

use of subheadings is appropriate transitional organizational technique

Yet there are problems with obtaining music. It is expensive, as composers often want large payments for well-crafted music. It is even more expensive if you want to hire real musicians with a multitude of instruments to play it, as opposed to a computer or a studio-engineer with a synthesizer. The money spent on this could be going towards the actual product, instead of supplements to the product. Thus companies are often discouraged from expanding on this important aspect, and either do it in a lackluster—sometimes outright detrimental—way, or not at all. They may only buy one or two songs, which are played constantly and annoy the user, who simply turns off the sound. Yet, if they could buy a simple, cheap program that would instantly compose new music each time it was accessed by their product, this would be much less of a problem and would provide the same benefits of more expensive music, while diverting less effort and financing away from their main focus. Also, a full score of music will take up much more space on both a computer and CD than a small program, which can pose minor complications or inconveniences.

sentence length + complexity not always appropriate for technical writing

The focus of this project is not only to build an effective virtual music composer, but also to test whether or not people would accept the music composed by a cheap, simple program. If it is shown that, overall, people have no problem with it, the project can be a building block for a corporate-level product. The program will be written in the Java programming language (using the Bluej Java interface) and Melody Assistant to play the program's output.

focuses on analytical questions within a technical piece

examples of technical language

organization – abstract, intro, results, discussion is appropriate for this kind of technical writing; however, the organization could be further adapted specifically for this piece

organization doe not provide a clear division to indicate process

Creating different classes made the organization and functionality of the program smoother. Outside of the main composer class, one was created to handle all of the information that pertained to a chord. Each chord used in the program was not a string of notes, but rather an instance of this class, within which the notes of the chord was a small part of information. It was within the Chord class that the method to randomly choose a next chord from its given list was stored. An abstract class called "phrases" was made, under which all other phrases (which were their own class) used as a template. Each class of a phrase knew how to create its own given phrase, returning the phrase when called (after it was given particular notes that it could start on, which was determined from the Chord class it would be played simultaneously with).

parenthetical explanations are not characteristic of technical writing

After the main "compose" method had been run, a .abc file was created. This file was properly formatted to be read by a dedicated music-program—In this case, Melody Assistant. Once opened with this program, individual notes could be seen and the composition could be exported to a music file with could in turn be played. For bug testing, a simple replacement of ".abc" with ".txt" in the file destination line would create a text file in ABC format which could be read.

The Survey

After the program was deemed complete—that is, once it could successfully output a functional piece of music—a single piece was composed and saved as an .mp3 file. Then, three other mainstream, human-composed pieces were rewritten in a manner that matches the output of the program: A chord progression underneath a single-note melody. Clips from the songs "My Father's Eyes" by Eric Clapton, "Hit That" by The Offspring, "Violin Concerto #1" by Paganini, and "We Didn't Start the Fire" by Billy Joel were used. It is important to note that these songs were chosen because they already matched the single-note-melody, basic-chord-progression style, and were thus not altered substantially by the transition. It was also important that they all be of a similar recording quality, so the computer-generated piece does not stand out so blatantly.

word choice jargon technical terminology is appropriate for genre

technical terminology used throughout suggests a specialized audience

Once all pieces were stored as .mp3 files, which could be played by any MP3 player (in this case, an iPod). They were presented in 30-second clips to participants (35 in total: 31 is a minimum for many statistical tests), who then answered 2 simple questions. They were asked to identify any piece that sounds unpleasant or annoying, and any that sounds "un-musical". Thus, after collecting the data, it can be formulated whether or not the output of the program was successful.

Using a random number generator, the experimental clip was assigned on the survey as "clip 1".

With the results, a 1-proportion z-test was used, using a TI-84+ graphing calculator (although it will work with any graphing calculator at least as new as the TI-83).

Results

example of less effective word choice for technical study

The program was completed successfully. Although errors and snags were encountered, it was negligible and did not prevent the desired result from being achieved. A functional mp3 file was exported, and the other songs were transposed to simpler format as well.

specialized reader understands/ can use the information

The results of the survey were as follows:

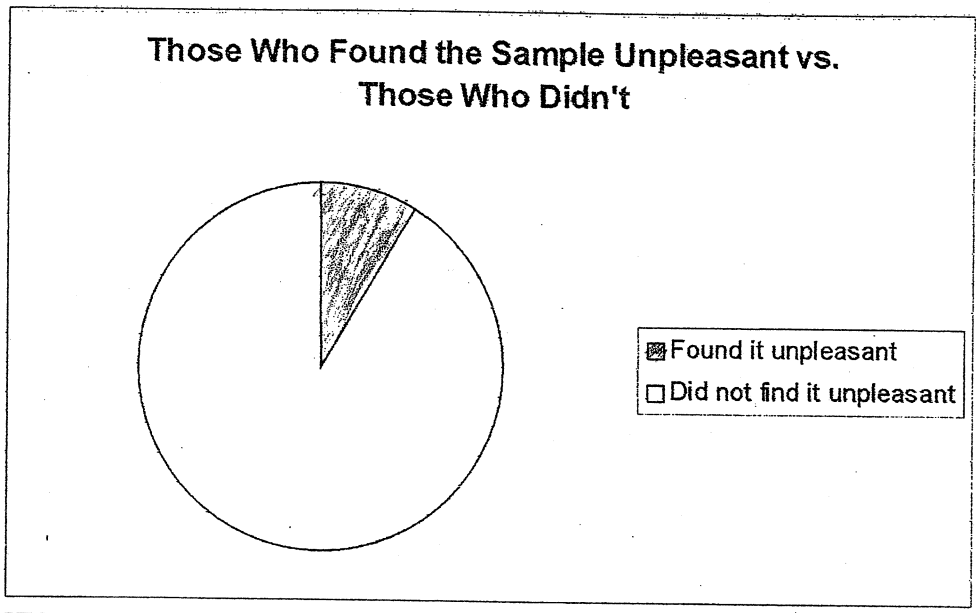
For question one, 2 out of the 35 applicants found the generated clip to be annoying or unpleasant. Question two yielded 12 who found it un-musical. That's 5.71% who found it unpleasant and 34.29% who found it un-musical.

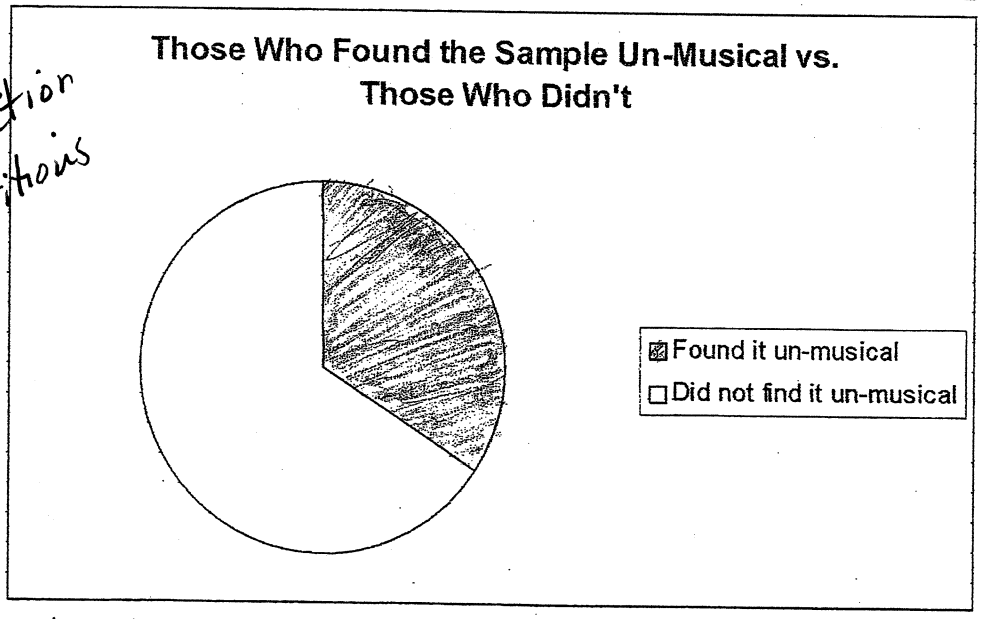
use of statistics as support

Using a graphing calculator, a 1-prop Z-test of the chance that less than a fourth of all people would find the results of the composer unpleasant resulted in a p-value of .0042. The same test yielded a p-value of .897 for the probability that less than a fourth would find it un-musical.

These p-values must be compared to the significance value chosen earlier. If lower, the notion of less than 25% finding the piece either unpleasant or un-musical is rejected, meaning it is unlikely. Otherwise, it is accepted as being likely.

charts are often used to enhance meaning; however these charts report information from the writing- placement is not near written explanation - information seems repetitious





Found it unpleasant	2
Did not find it unpleasant	32

writing demonstrates control of grammar/usage, and correctness evidenced by absence of errors.

Found it un-musical	12
Did not find it un-musical	23

Discussion

When checked against the significance value of .1, the p-value of the first question is much lower, meaning that it is very likely that less than 25% of all people find the results of the program unpleasant. The p-value of the second question, however, was much higher than the significance value, meaning that it is very unlikely that 25% or less of the population would find the piece unmusical. Therefore, half of the hypothesis was correct.

idea development is analytical and insightful throughout

The purpose of this experiment was to determine if it was likely that a simple computer composer could take the place of human composers for simple background music in commercial products. If this is to be true, it is absolutely imperative that a virtually-composed piece is not unpleasant to the listener. Not only will it obviously produce discomfort, but may do it to the extent that the consumer will be unsatisfied with the product. While good music may provide a great deal of immersion value for games, bad music can make immersion very difficult. It can disrupt thinking. It is better to have nothing at all than a negative thing. However, this thankfully does not seem to be the case, as it was found very likely that less than a fourth of all people will find it unpleasant. Although a fourth is still a very large portion realistically—especially for a commercial product—for a "rough-draft" of a program, it is suffice to prove that it is very likely for a program to produce pleasant music.

Discussion section concludes analysis

Musicianship is also paramount to the success of good music. There is more to music than just playing notes that sound good together—if it lacks certain character, melody, or feeling, people will not be able to connect with it. Unfortunately, this experiment was not able to prove that a simple program could produce something melodious or otherwise "musical". It is therefore unlikely that such a program could take the place of a main theme song, and certainly not a mainstream stand-alone song.

ideas are developed in detail

Although most of the project went off smoothly, some things could have been done more effectively. Vague planning created a few problems and time-sinks when coding the program. If this or a similar project is to be repeated, it will definitely have a more solid plan beneath it before any coding is done. The survey, while mostly effective, could have been more comprehensive and set-up to have more informative results. As it is, all that is learned from the 2nd question (which did not 'pass') is that the portion of the population that would think the piece unmusical is not 25% or lower. The high p-value indicates only that it is very unlikely, not what a more realistic proportion would be (which would be useful to know). The nature of these results makes it difficult to compare the test to other similar tests. Also, the manner in which this experiment could be carried out could vary drastically, meaning that a viable match to compare it with would be difficult to find.

sentence structure is varied and effective though perhaps more developed than usual in technical writing

Overall though, it is believed that the experiment as a whole was a success in its goal. The future of a computer composer is not to produce main theme songs or mainstream radio hits, but to generate backround "filler" music. As long as the music is not unpleasant (which the experiment showed it was not), then it doesn't have to be melodious. In fact, melodious music may distract from the desired focus of the product more than other pieces. Judging by this project, a virtual composer seems to have a definite chance of success.

Bibliography

Aloupis, Greg, Thomas Fevens, Tomomi Matsui, Stefan Langerman, Yurai Nunez, Antonio Mesa, David Rappaport, and Godfried Toussaint. "· Algorithms for Computing Geometric Measures of Melodic Similarity." Computer Music Journal 30 (2006): 67-76. School of Computer Science, Louisville. 6 Oct. 2006.

Blood, Brian. "Music Theory & History Online." Dolmetsch Online. 8 Oct. 2006 <http://www.dolmetsch.com/theoryintro.htm>.

Eachus, Peter. "Survey Techniques." University of Salford. University of Salford. 8 Oct. 2006 <http://www.chssc.salford.ac.uk/healthSci/resmeth2000/resmeth/ch2surv.htm>.

Hanninen, Dora A. "· a Theory of Recontextualization in Music: Analyzing Phenomenal Transformations of Reptition." Music Theory Spectrum (2003): 1-39. Research Library, Louisville. 23 Sept. 2006.

Hewlett, Walter B., and Eleanor Selfridge-Field. "Directory of Computer Assisted Research in Musicology." Computing in Musicology 3 (1987): 1-76. Center for Computer Assisted Research in Humanities. 23 Sept. 2006.

McClelland, Ryan. "· Extended Upbeats in the Classical Minuet: Interactions with Hypermeter and Phrase Structure." Music Theory Spectrum 1 (2006): 1-34. Louisville. 23 Sept. 2006.

Narins, Pamela N. "Survey Techniques." Ryerson. Ryerson University. 8 Oct. 2006 <http://www.ryerson.ca/~mjoppe/ResearchProcess/SurveyTechniques.htm>.

Walshaw, Chris. "The Abc Musical Notation Language." The Abc Musical Notation Language. 12 Oct. 2006 <http://abcnotation.org.uk/>. 1 July 2006.

Whitesell, Lloyd. "· Speech-Melody and Primitive Music." Ed. Leon Botstein. The Music Quarterly 88 (2005): 167-203. 23 Sept. 2006.

Zimmerman, Paul, comp. "Chord Theory." Harmony Central. 23 Sept. 2006 <http://www.harmony-central.com/Guitar/chord-and-theory-chart.txt>.

extensive bibliography is common in scientific studies

Rationale
"The Virtual Composer"
Analytical/Technical Sample
Grade 12
2009

__4___ Content (4, 4, 4)

Using a real question from the field allows the student to establish an authentic purpose. The writer maintains the focus throughout the piece. Insight is evident in the connections made with real-world applications and within the process itself. The audience is specific, and the writer communicates effectively with the audience throughout the piece, both in awareness of audience's needs and awareness of audience's prior knowledge. The tone of the piece is appropriate for the genre. Idea development is thorough and specific, with careful attention to detail and without bias.
Charts (or other graphics) can be used to enhance meaning; however, the charts in this piece merely repeat information from the text. Additionally, these charts are not inserted at the appropriate place and they are over-sized.

__3___ Structure (3, 3, 3)

While the piece demonstrates logical, coherent organization, and the piece reflects careful and purposeful organization, the text-heaviness of the piece, along with the generic/formulaic use of subheadings, precludes its being as effective as it might have been. The use of subheadings as a transitional/organizational technique is appropriate to the genre; however, the subheadings used here are formulaic and generic, and their use would have been more effective if they had been adapted more specifically for this paper. The use of sentences of greater length and complexity, which would be valued in pieces from other genres, is not appropriate here (e.g., "Yet, if they could buy a simple, cheap program that would instantly compose new music each time it was accessed by their product, this would be much less of a problem and would provide the same benefits of more expensive music, while diverting less effort and financing away from their main focus.")

__4___ Conventions (4, 4, 3)

The language in the piece is precise and technically correct. Technical terms (e.g., "random number generator," "single-note-melody, basic-chord-progression," "1-proportion z-test") are used effectively. Passive voice, appropriate for this genre, is used throughout, which helps maintain the tone of the piece. The use of parentheses throughout the Discussion section does not meet the standards for correctness for the genre (e.g. "which did not pass," "which would be useful to know") as it detracts from the tone of the piece: if it is important enough to include, do not use parentheses.